(72) Inventors:
     • Zenchelsky, Daniel N.
       San Jose, CA 95123 (US)

     • Dutta, Partha P.
       San Jose, CA 95129 (US)
     • London, Thomas B.
       Mountain View, CA 94040 (US)
     • Vrsalovic, Dalibor F.
       Sunnyvale, CA 94087 (US)
     • Sül, Karl Andres
       Princeton, New Jersey 08540 (US)

(74) Representative:
     Modiano, Guido, Dr.-Ing. et al
     Modiano, Josif, Pisanty & Staub,
     Baaderstrasse 3
     80469 München (DE)

(54)   **System and method for providing peer level access control on a network**

(57)   A system and method for providing peer-level
access control on networks that carry packets of infor-
mation, each packet having a 5-tuple having a source
and destination address, a source and destination port,
and a protocol identifier. The local rule base of a peer is
dynamically loaded into a filter when the peer is authen-
ticated, and ejected when the peer is loses authentica-
tion. The local rule base is efficiently searched through
the use of hash tables wherein a hashed peer network
address serves as a pointer the peer's local rules. Each
rule comprises a 5-tuple and an action. The action of a
rule is carried out on a packet when the 5-tuple of the
rule corresponds to the 5-tuple of the packet.

FIG. 9

EP 0 854 621 A1
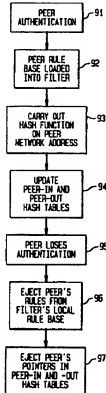
Description

Field of the Invention

5   This invention relates to information systems security, in particular to providing access control between one set of automated information systems and another.

Background of the Invention

10   Known methods for implementing access control for a specific computer on a network are cumbersome and inflexible because access rules must be coded and entered by hand by a system administrator. This is impractical for networks whose members change frequently, or whose members' security needs change frequently.

Effective information systems security prevents the unauthorized disclosure, modification or execution of an automated information system's (AIS) data and processes. As used here, the term AIS refers to a computer, network of
15   computers, internetwork of computers, or any subset thereof. The term "data" refers to any information resident on an AIS, including files and programs. The term "processes" refers to programs in any stage of execution on an AIS.

A "host" is a computer with an assigned network address, e.g., an Internet Protocol (IP) address. A "user" is a computer that does not have a fixed, assigned network address. To obtain connectivity to the Internet, for example, a user must commonly obtain a temporary IP address from a host with a pool of such addresses. Such a temporary IP address
20   is retained by the user only for the duration of a single session of connectivity with the Internet.

Information flows in certain networks in packets. A "packet" is a quantum of information that that has a header containing a source and a destination address. An example of a packet is an IP packet. Packets such as IP packets have a network protocol identifier ("protocol") as a part of packet header. The protocol identifies the version number of the protocol used to route the packet. An example of a network protocol identifier is the IP protocol field in an IP packet
25   header.

Packets on a network are directed to and from ports. A "port" is a logical address within a computer through which a process executing on the computer communicates with other executing processes. These other processes may reside on the same computer, or on other networked computers.

Information systems security is implemented by means of a security policy, which comprises rules directed towards
30   regulating the flow of information in an AIS. The rules of a security policy are embodied in a "rule base," a set of rules that specify whether a packet should be passed to the intended recipient or dropped based upon the packet's identifier. A packet identifier is data generally carried in the packet header that serves to identify the packet. An example of a packet identifier is a circuit number, which occurs in the headers of packets flowing in connection-oriented (i.e., circuit-switched) packet switched networks. Another example of a packet identifier is a packet 5-tuple, which is the packet's
35   source and destination address, source and destination port, and protocol. Packets with 5-tuples flow in connectionless packet switched networks.

A rule base may be global or local. A global rule base is a uniform set of rules ("global rules") that apply to a group of users, hosts, or both. A local rule base is a set of rules ("local rules") that apply to a single user with a temporary network address or a host. A single user with a temporary network address or a host that has its own rule base is called
40   a "peer."

Another means for implementing security policy is to restrict access to a network to a predetermined set of users and hosts. When a user or host requests access, its identity must be established and verified before access is granted. This process implicates two steps: identification and authentication.

FIG 1 shows one method of identification and authentication in the form of a flow chart with each step designated
45   by a reference numeral. A first step requires a source of information to identify itself by name by supplying a string of data called a user id 10. To prevent an imposter from obtaining the privileges associated with a given user id, the user behind the user id is verified by requiring it to provide a password 11 that is normally kept confidential. Such verification is called "authentication." The AIS checks the combination of source id and password against a list of valid users, 12. When the AIS recognizes a valid user id and corresponding password, a user or host is said to have been identified and
50   authenticated 14. Otherwise, the request for access is denied 13. Hereinafter, a source that has been identified and authenticated will be said to have been "authenticated" for purposes of brevity.

A security policy rule base is implemented on a network using a device called a filter comprising hardware and software. The rule base is loaded into the filter, which receives packets en route (between their source and destination) and checks the identifier of each packet against the identifier contained in each rule of the rule base for a match, i.e., if the
55   packet corresponds to the rule. A packet corresponds to a rule if the rule applies to the packet. Hence, a rule that is meant to apply to packets with a circuit number of 3254, for example, "corresponds" to all packets with a packet identifier that indicates circuit number 3254. If the network packet identifier corresponds to a rule identifier, the filter carries out the PASS or DROP action prescribed by the rule on the packet. If the PASS action is carried out, the packet is

allowed to pass through the filter. If the DROP action is carried out, the packet is eliminated.

A filter is often combined with other hardware and software that helps manage the flow of information through the filter. The combination of hardware and software that carries out and supports packet filtering is called a firewall. A firewall is often positioned between a first network that "owns" the firewall and a second network. The purpose of the firewall is to regulate the flow of information into and out of the first network from the second network by implementing the rule base belonging to the first network for all such information.

A typical application of a firewall is shown in FIG 2. A corporate network 20 may wish to provide access to Internet hosts 21 to its subscribers, but may wish to limit the access that the Internet hosts 21 have to the corporate network 20, which may contain trade secrets and proprietary information. The corporate network 20 would develop a security policy implemented by a firewall 22 placed at the interface between the corporate network 20 and the Internet hosts 21. The firewall 22 comprises a filter 23 that would PASS or DROP packets from Internet hosts 21 to corporate network subscribers 20 and vice versa based upon the packets' source and destination addresses. The firewall is said to belong to the corporate network, and enforces rules that "protect" hosts within the corporate network that have IP addresses. Such hosts are said to be "behind" the corporate network firewall.

An example of a rule base for corporate network 20 having hosts A 24, B 25 and C 26, connected through a firewall 22 to the Internet having hosts G 27, H 28 and I 29 is as follows:

SOURCE Address, Port	DESTINATION Address, Port	VERSION	ACTION
A,21	G,32	4	PASS
A,22	H,19	3	DROP
G,11	A,64	4	DROP
C,9	I,23	4	PASS

Every rule base must also have a default action for transactions that are not explicitly specified in the rule base, which is usually the DROP action. Thus, packets from system A,21 to system G,33 will be dropped because the above rule base does not expressly include a rule for such a transfer.

A typical architecture for providing users access to the Internet is shown in FIG 3. Users 31 and 32 do not have fixed IP addresses. Rather, a user is assigned temporary IP addresses by an Internet Service Provider (ISP) Point of Presence (POP) 33 from a pool of addresses kept by the POP 33 for this purpose. A POP comprises at least one host (not shown). When a user 31 terminates his session of access to the Internet 35, the IP address is returned to the POP 33. Thus, over successive access sessions, a user 31 is likely to have several different IP addesses.

Known filters are not well suited to providing appropriate access control for networks such as a POP. This is because a known filter is only able to load and store rules through the intervention of a system administrator, a slow and cumbersome process. Indeed, the system administrator generally must hand-code rules in a format specific to the filter platform. With known filters, it is impractical to implement the access rules of a specific user (known as the user's "local rules") who is accessing and leaving the network with changing network addresses.

This problem is illustrated in FIGs 5a and 5b. FIG 5a shows a first session where a first user 51 has requested Internet access and been authenticated by a POP and been assigned IP address B from the POP IP address pool 52. Likewise, a second user 53 has been authenticated and been assigned IP address E from the pool 52. A rule base 53 is loaded into a filter to regulate the flow of information between users 51 and 53 and the hosts P, U, V and W on the Internet. The rule base shown in FIGs 5a and 5b show only the source and destination addresses for each rule, and omit source and destination ports and protocol for simplicity.

Both users stop accessing the Internet and then later request access again and are authenticated for a second session, shown in FIG 5b. This time, the first user 51 is assigned IP address E from the pool 52, and the second user is assigned IP address A. With the newly assigned network addresses, the rule base in the filter is now out of date, containing no rules for the second user, and the wrong rules for the first user, which has been assigned the IP address assigned to the second user during the first session. Even if both users had fortuitously been reassigned the same IP addresses for their second sessions, if either user's security needs had changed between sessions, a new rule base would have had to be loaded into the filter. As discussed above, loading rules into known filters is tedious. Loading and dropping such rules with the frequency that users access and leave a POP is impractical for known filters.

The inflexibility of known filters often necessitates the implementation of rule bases that are too broad for a given application. Without the possibility of easy updates, it is simpler to mandate global rules that apply to all AIS behind a

filter rather than to load rules that apply to specific hosts. In such a case, all AIS behind the filter must conform to the most restrictive security requirements of any such AIS, resulting in overly restrictive filtering.

The shortcomings of known filters are illustrated by some of the architectures presently used to provide information systems security for a POP. The architecture shown in FIG 3 provides a minimal level of security through an authentication system 34 which limits access to a predetermined list of authenticated users. But the list of users must generally be entered by hand by the system administrator, and so cannot be easily changed. Further, once access is granted, the access is unlimited. Information may flow to and from users 31 and 32 from the Internet 35 without regulation, providing no security past the initial authentication process. This exposes users 31 and 32 to the risk of hacker attacks from users and hosts on the Internet, possibly resulting in the theft or unauthorized manipulation of user data.

The architecture illustrated in FIG 4 shows another known solution to providing information systems security on a POP. The known filter 46 implements a security policy for packets flowing between the Internet 45 and hosts 41 and 42. However, the rule base in the filter 46 must still be formulated and loaded by the system administrator. Further, the network addresses of the users 31 and 32 are likely to change on a session by session basis. This means that it is only practical to load general, "global" rules into the filter that are valid for all of the users. Thus, for example, if user A does not wish to receive packets from a particular host on the Internet, the filter rule base must drop all such packets, thus cutting off user B from receiving packets from that Internet host as well. In this way, the global rule base necessitated by the limited capabilities of known filtering systems is almost always too broad. Another disadvantage is that it is difficult to change the filter rule base to accommodate changing security needs of either user 41 or 42.

Another architecture that provides security on for each peer is shown in FIG 6. Here, filters 66 and 67 are placed between users 61 and 62, respectively, and the POP. Requiring every user to have its own filter is an expensive solution that is impractical to implement.

What is needed is a filtering system and method that accurately and efficiently implements local rule bases on a network whose configuration and security needs are constantly changing. Such an invention would provide peer-level security flexibly and inexpensively, with little intervention required from a system administrator.

Summary of the Invention

The present invention comprises a filter that efficiently stores, implements and maintains access rules specific to an individual computer on a network with rapidly changing configurations and security needs. This advantageously allows an individual computer (a peer) to implement its security policy on a filter shared by many such computers on a network.

When a local rule base is no longer valid because the peer is no longer authenticated to the filter in accordance with the present invention, the peer's local rule base is "ejected," i.e., a logical operation is carried out at the filter whereby the local rule base is deleted from the filter. This logical operation of stored data in a computer is well known in the art. This effectively regulates the flow of information on session-by-session basis, which is especially advantageous in AIS where individual users and hosts have different security needs that change from time to time. For example, the present invention is useful for implementing a parental control system wherein a parent is able to regulate the access to certain types of licentious material on the Internet for household Internet access accounts.

The present invention allows a single device to flexibly and efficiently regulate the flow of information in accordance with security policies that are specifically tailored to the individual user or host. Advantageously, no intervention on the part of the system administrator is ordinarily required in the ordinary functioning of the present invention. Unlike known filters, the present invention is able to accommodate users with temporary network addresses as easily as hosts with fixed network addresses.

In accordance with the present invention, each individual peer is authenticated upon requesting network access. The peer's local rule base is then loaded into the filter of the present invention, either from the peer itself, or from another user, host or peer. When the peer is no longer authenticated to the POP (e.g., the peer loses connectivity or logs off from the POP), the peer's local rule base is ejected (deleted)from the filter.

Brief Description of the Drawings

FIG 1   shows the process of identification and authentication.
FIG 2   shows a firewall interposed between a corporate network and the Internet.
FIG 3   shows users connected to the Internet through a Point of Presence (POP) having an authentication system.
FIG 4   shows a POP with an authentication system and a filter.
FIG 5a   shows a first Internet access session for two users through a POP having a filter.
FIG 5b   shows a second Internet access session for two users through a POP having a filter.
FIG 6   shows a known method of providing user level access control to the Internet.
FIG 7a   shows a rule base architecture in accordance with an embodiment of the present invention.

Detailed Description

In accordance with the present invention, FIG 7a shows an embodiment of a rule architecture that incorporates the functionality of known filters by including a global pre-rule base 701, a local rule base 702 and a global post-rule base 703.

The global pre-rule base 701 usually comprises general rules that apply to all hosts behind the firewall, and are most efficiently applied before any local rules. An example of a global pre-rule is that no telnet (remote login) requests are allowed past the firewall.

The local rule base 702 comprises the set of peer rule bases loaded into the filter for authenticated peers. These rule pertain to specific hosts. An example of a local rule is that host A may not receive e-mail from beyond the firewall.

The global post-rule base 703 comprises general rules that are most efficiently applied after the global pre-rule base and local rule base is searched. A rule applied in the global post-rule base need not have the same effect as if it were applied in the global pre-rule base. Consider the above example prohibiting the reception of certain telnet requests. If this rule is placed in the global post-rule base, the local rule base is searched first, and may contain a rule allowing a telnet request through for a particular peer. If such a rule is found in the local rule base, the global post-rule base is not subsequently searched, and the telnet request is allowed to pass. Consider the different effect of the same rule when it occurs in the global pre-rule base, which is to block all telnet requests for all hosts behind the firewall. The importance of the order of applying rules is evident from a more thorough consideration of the method of the present invention.

FIG 7b illustrates a flow chart of packet processing or filtering in accordance with the present invention. As shown therein, a packet entering the filter is first checked against a global pre-rule base 711 containing rules for all hosts and users having network addresses behind the firewall.

If a corresponding rule is found and the prescribed action is DROP, the packet is dropped 712. If a corresponding rule is found and the action is PASS, the packet is passed 720. If no corresponding rule is found, then the local rule base is checked 713.

The local rule base 702 is the set of all per user rule bases that are dynamically loaded upon authentication and ejected upon loss of authentication in accordance with the present invention.

If a corresponding rule is found in the local rule base and the action is DROP, the packet is dropped 714. If a corresponding rule is found and the action is PASS, the packet is passed 721. If no corresponding rule is found, then the global post-rule base is checked 715.

If a corresponding rule is found in the global post-rule base and the action is DROP, the packet is dropped 716. If the action is PASS, the packet is passed 722. If no corresponding rule was found in any of the rule bases, then the packet is checked against the default rule 717, whose action is generally to DROP the packet. If the packet corresponds to the default rule, then the default action is carried out 723. If the packet does not match the default rule, then an error condition occurs 724.

This rule base architecture advantageously retains the functionality of known filters. For example, if there are rules in the global pre- or post-rule base only, the filter behaves the same as known filters. If there are only rules in the local rule base, the filter has all of the new and innovative features of the present invention without having global rules.

It is advantageous to implement the present invention with a system for efficiently searching the local rule base for corresponding rules for a given packet. A system that provides such efficiencies uses a hash function to generate an index for the rules. A hash function maps a string of characters to an integer. As is known in the art, a character string is represented as binary numbers inside a computer. An example of a hash function would be to take the third, fourth and fifth bytes of a character string as it is stored in a computer as the first, second and third digits of an integer to be associated with the string. A string on which a hash function has been carried out is said to be "hashed," and the resulting integer is referred to as the "hash" of the string.

This is carried out by logically dividing the local rules into local-in rules and local-out rules. A local-in rule is any rule that applies to a packet whose destination address corresponds to a network address behind the firewall. For example, suppose a host with network address A is behind the firewall, and hosts B, C and D are outside the firewall. The following are examples of local-in rules for host A, following the format SOURCE ADDRESS, SOURCE PORT--> DESTINATION ADDRESS, DESTINATION PORT: Protocol: ACTION:

B,31-->A,33:4:DROP
C,64-->A,45:4:PASS
D,11-->A,17:4:PASS

A local-out rule is any rule that applies to a packet whose source corresponds to a network address behind the firewall. Local out-rules for the above example are:

A,44-->B,70:4:PASS
A,13-->C,64:4:DROP
A,12-->D,17:4:DROP

In accordance with the present invention, a hash function h is carried out on the network address of the owner of a local rule base. A hash function associates an integer with a string. For the above example in which a host with network address A ("host A") has a local rule base, a hash function would be carried out on A: h(A)=N , where N is an integer

An example of such a hash function is to take the last decimal digit in each octet of an IP address and compose an integer for the hash number. Thus, for example, the IP address 123.4.46.135 would have a hash value of 3465.

After the hash function is carried out, a local-in and a local-out hash table is generated. These tables are essentially indexes searchable on hash numbers derived from network addresses of peers, where each hashed peer network address points to that peer's local-in and local-out rules. Thus, if A is the network address of peer A, and if h(A)=32, then 32 would point to peer A's local-in and local-out rules in the local rule base.

The advantages of this indexing system in accordance with the present invention may be demonstrated with the aid of FIGs 8a, 8b, 8c and 8d. FIG 8a shows an example architecture where peers A 801, B 802, and C 803 are behind a firewall 804 having a filter 805 connected to a network 806 having hosts G 807, H 808 and I 809. These letters represent network addresses. FIG 8b shows the local rule base associated with each host. For simplicity, each rule in the rule bases is shown only as a network source and destination address; the source and destination ports and protocol numbers are not shown. The asterisk represents a wildcard indicating any host. For example, this feature may be advantageously implemented in accordance with the present invention by including wildcards in one or more of the four octets that constitute an IP address. The following IP address specifications are all valid for use in rule bases in accordance with the present invention:

123.*.233.2
34.*.*.155
..*.32
..*.*

The wildcard feature may also be used in accordance with the present invention in a similar fashion in any other component in the 5-tuple, i.e., the source and destination ports and the protocol.

FIG 8c shows the peer-in hash table 821 and peer-out hash table 822 derived from the local rules shown in FIG 8b and hash function h carried out on network addresses A, B and C 823. When a packet is received by the filter 805, the filter carries out the same hash function h on the packet's source and destination address 824.

FIG 8d shows the method by which the hash tables are searched in accordance with the present invention. FIG 8d represents a detailed view of the box "Check Local Rule Base" 713 in FIG 7b.

In accordance with the present invention, if there was no corresponding rule found in the global pre-rule base 711 (FIG 7b), then the local-in hash table is efficiently searched for a rule that corresponds to the packet 841. If a corresponding rule is found and the action is DROP, the packet is dropped 842. If the action is PASS or there is no corresponding rule, the peer-out hash table is checked 843. If a corresponding rule in the hash-out table is found and the action is DROP, the packet is dropped 844. If the action is PASS or there is no corresponding rule, and if at least one of the hash tables contained a corresponding rule, the packet is passed 845. If there were no corresponding rules in either hash table 846, then the post-rule base is checked 715 as shown in FIG 7b.

Were it not for the peer-in and peer-out hash tables, the rules would have to be searched far less efficiently by searching the entire rule base for rule identifiers (e.g., 5-tuples) that match the packet identifier (e.g., 5-tuple.) The part of the rule that identifies the packet to which the rule applies (the rule identifier) is also called the rule "key." Using hash tables eliminates the need to search the keys of all rules, pointing instead to the relevant subset of possibly applicable rules through a speedier search. Thus, the scope and computational time needed to carry out the search is substantially and advantageously reduced, reducing the delay in packet transit time caused by the interposition of a filter between the packet source and destination.

As shown in FIG 9, a peer is first authenticated 91 in accordance with the present invention. Upon authentication, the peer's local rule base is loaded into the filter 92. A hash function is carried out on the peer's network address 93,

and the filter's peer-in and peer-out hash tables are updated 94 with pointers to the peer's peer-in and peer-out rules. When the peer is no longer authenticated 95, the peer's local rules are ejected from the filter local rule base 96, and the pointers to the peer's peer-in and peer-out rules are ejected from filter's peer-in and peer-out hash tables 97.

The present invention provides new security functionality on a per user basis on filters and firewalls, while maintaining the functionality of known filters. The present invention allows for the dynamic adjustment of local rule bases that can be dynamically tailored to meet the changing needs of the individual user.

Where technical features mentioned in any claim are followed by reference signs, those reference signs have been included for the sole purpose of increasing the intelligibility of the claims and accordingly, such reference signs do not have any limiting effect on the scope of each element identified by way of example by such reference signs.

Claims

1. A filter for providing peer level access control on a network having a peer with a local rule base, wherein said filter comprises:

   a. means for accessing a peer's local rule base; and
   b. means for receiving a packet having a packet identifier, identifying a corresponding local rule, and carrying out the action of the corresponding local rule on the packet while said filter is filtering packets for the peer. on the packet.

2. The filter of claim 1, further comprising:

   c. means for ejecting said local rule base from said filter.

3. The filter of claim 1, wherein the packet identifier comprises a source and destination address, a source and destination port, and a protocol identifier.

4. The filter of claim 1, wherein said means for accessing the local rule base comprises receiving and storing the local rule base.

5. The filter of claim 1, further comprising means for authenticating the peer.

6. The filter of claim 1, further comprising a global pre-rule base having a global pre-rule, wherein upon receiving the packet, said filter first searches said global pre-rule base for a rule that corresponds to the packet and carries out the action of the corresponding global pre-rule on the packet, and wherein if no corresponding global pre-rule is identified, the filter searches the local rule base for a rule that corresponds to the packet and carries out the action of the corresponding local rule on the packet.

7. The filter of claim 1, further comprising a global post-rule base, wherein the global post-rule base is searched for a rule that corresponds to the packet, and the action of a global post-rule is carried out if it corresponds to the packet only if no corresponding rule in said global pre-rule base and no corresponding rule in said local rule base are identified.

8. The filter of claim 1, further comprising a default rule, wherein if no corresponding pre-global rule and no corresponding local rule and no corresponding post-global rule are identified, said filter carries out the action of said default rule if said default rule corresponds to the packet, and generates an error condition if said default rule does not correspond to the packet.

9. The filter of claim 1, wherein the peer has a network address, and wherein the packet identifier comprises a packet source address and a packet destination address, and wherein a local rule comprises a rule source address, a rule destination address and an action, further comprising a local-in hash table having an in-pointer derived by applying a hash function to the network address of the peer, said in-pointer pointing to a peer's local rule whose rule destination address corresponds to the network address of said peer.

10. The filter of claim 1, wherein the peer has a network address, and wherein the packet identifier comprises a packet source address and a packet destination address, and wherein a local rule comprises a rule source address, a rule destination address and an action, further comprising a local-out hash table having an out-pointer derived by applying a hash function to the network address of the peer, said out-pointer pointing to a peer's local rule whose rule

source address corresponds to the network address of said peer.

11. A method for providing peer-level access control on a network, said method comprising:

     a. accessing a local rule base of a peer;
     b. receiving a packet having a packet identifier; and
     c. searching the local rule base, identifying a local rule that corresponds to the packet identifier, and carrying out the action of a local rule if the local rule corresponds to the packet.

12. The method of claim 11, further comprising the step of:

     d. ejecting the local rule base.

13. The method of claim 11, wherein the packet identifier comprises a source and destination address, a source and destination port, and a protocol identifier.

14. The method of claim 11, wherein accessing a local rule base comprises the steps of receiving and storing the local rule base.

15. The method of claim 11, further comprising step of authenticating a peer before accessing the peer's local rule base.

16. A method for providing peer-level access control on a network with a peer, said method comprising:

     a. receiving a packet having a packet identifier;
     b. searching a global pre-rule base and identifying a global pre-rule that corresponds to the packet;
     c. carrying out the action of a global pre-rule if the global pre-rule corresponds to the packet;
     d. accessing a local rule base of a peer;
     e. if no corresponding global pre-rule is found in the global pre-rule base, searching the local rule base, identifying a local rule that corresponds to the packet, and carrying out the action of a local rule if the local rule corresponds to the packet.

17. The method of claim 16, further comprising the step of:

     f. ejecting the local rule base from the filter.

18. The method of claim 17, further comprising the steps of:

     g. if no corresponding global pre-rule is found in said global pre-rule base and no corresponding local rule is found in said local rule base, searching a global post-rule base for a global post-rule that corresponds to the packet; and
     h. carrying out the action of a global post-rule if the global post-rule corresponds to the packet.

19. The method of claim 18, further comprising the steps of:

     i. if no corresponding rule is found in the global pre-rule base and no corresponding rule is found in the local rule base, and no corresponding rule is found in the global poet-rule base, determining if the packet corresponds to a default rule; and
     j. carrying out the action of the default rule if the default rule corresponds to the packet, and generating an error condition if the default rule does not correspond to the packet.

20. The method of claim 16, wherein the peer has a network address, and wherein the packet identifier comprises a packet source address and a packet destination address, and wherein a local rule comprises a rule source address, a rule destination address and an action, and wherein the local rule base having a local rule whose rule destination address corresponds to the network address of the peer is searched for a local rule that corresponds to the packet, and the action of a local rule is carried out if the local rule corresponds to the packet; comprising the steps of:

a. deriving an in-pointer by applying a hash function to the network address of the peer;

b. storing the in-pointer in a peer-in hash table such that the in-pointer points to a local rule whose rule destination address corresponds to the network destination address of the peer;

c. receiving a packet;

d. applying the hash function to the network destination address of the packet;

e. searching the local rules to which the in-pointer corresponding to the hashed packet network destination address points for a rule that corresponds to the packet; and

f. carrying out the action of a rule if the rule corresponds to the packet.

21. The method of claim 20, further comprising the step of:

      g. deleting the peer's in-pointers in said local-in hash table.

22. The method of claim 16, wherein the peer has a network address, and wherein the packet identifier comprises a packet source address and a packet destination address, and wherein a local rule comprises a rule source address, a rule destination address and an action, and wherein the local rule base having a local rule whose rule source address corresponds to the network address of the peer is searched for a local rule that corresponds to the packet, and the action of a local rule is carried out if the local rule corresponds to the packet, comprising the steps of:

      a. deriving an out-pointer by applying a hash function to the network address of the peer;

      b. storing the out-pointer in a peer-out hash table such that the out-pointer points to a local rule whose rule source address corresponds to the network source address of the peer;

      c. receiving a packet;

      d. applying the hash function to the network source address of the packet;

      e. searching the local rules to which the out-pointer corresponding to the hashed packet network source address points for a rule that corresponds to the packet; and

      f. carrying out the action of a rule if the rule corresponds to the packet.

23. The method of claim 22, further comprising the step of:

      g. deleting the peer's out-pointers in said local-out hash table.

24. A filter for providing peer-level access control on a network with a peer, said filter comprising:

      a. means for authenticating a peer;

      b. means for accessing rules from a peer that prescribe a PASS or DROP action to be carried out on a packet;

      c. means for receiving a packet;

      d. means for searching for and identifying rules that match the packet; and

      e. means for carrying out the PASS or DROP action of a rule that corresponds to the packet.

25. The filter of claim 23, further comprising:

      f. means for ejecting the rules of a peer.

26. The filter of claim 24, further comprising:

      g. means for authenticating a peer.

FIG. 1

(PRIOR ART)

```
┌──────────┐
│ OBTAIN   │ ── 10
│ USER ID. │
└──────────┘
     │
     ▼
┌──────────┐
│ OBTAIN   │ ── 11
│ PASSWORD │
└──────────┘
     │
     ▼
     ◇ SOURCE               13
    ╱ ID AND PASSWORD ╲  NO  ┌─────────┐
    ╲ CORRESPOND?     ╱ ────▶│ REJECT  │
     ◇                      │ SOURCE  │
     │ YES                  └─────────┘
     ▼
┌────────────────────┐
│ SOURCE IDENTIFIED  │ ── 12
│ AND AUTHENTICATED  │
└────────────────────┘
```

FIG. 2

(PRIOR ART)

FIG. 3
(PRIOR ART)

AUTHENTICATION SYSTEM — 34

31 — USER A

32 — USER B

POP — 33

INTERNET — 35

FIG. 4
(PRIOR ART)

AUTHENTICATION SYSTEM — 44

41 — USER A

42 — USER B

FILTER — 43

46

POP

INTERNET — 45

FIG. 5A
(PRIOR ART)

POP IP ADDRESS POOL	SESSION 1	FILTER RULE BASE

```
    POP IP                SESSION 1
ADDRESS POOL                                FILTER RULE BASE

       A        (FIRST                        B ──U   PASS
       B ─── USER) B                          B ──V   DROP
       C        (SECOND                       P ──B   DROP
       D ─── USER) E
       E                                      E ──V   DROP
       F                                      E ──W   DROP
                                              W ──E   PASS
```

FIG. 5B
(PRIOR ART)

```
    POP IP                SESSION 2
ADDRESS POOL                                FILTER RULE BASE

       A        (FIRST                        B ──U   PASS
       B      USER) E                         B ──V   DROP
       C        (SECOND                       P ──B   DROP
       D      USER) A
       E                                      E ──V   DROP
       F                                      E ──W   DROP
                                              W ──E   PASS
```

FIG. 6
(PRIOR ART)

12

FIG. 7A

GLOBAL PRE-RULE BASE ~701
LOCAL RULE BASE ~702
GLOBAL POST-RULE BASE ~703

FIG. 7B

```
                              ┌711
┌─────────┐         ╱─────────╲              ┌712
│  PASS   │  PASS  ╱   CHECK    ╲    DROP    │  DROP   │
│ PACKET  │───────⟨ GLOBAL PRE-RULE⟩────────│ PACKET  │
└─────────┘        ╲    BASE?    ╱            └─────────┘
 720                ╲─────────╱
                         │ NO MATCH
                              ┌713
┌─────────┐         ╱─────────╲              ┌714
│  PASS   │  PASS  ╱   CHECK    ╲    DROP    │  DROP   │
│ PACKET  │───────⟨  LOCAL RULE  ⟩────────│ PACKET  │
└─────────┘        ╲    BASE?    ╱            └─────────┘
 721                ╲─────────╱
                         │ NO MATCH
                              ┌715
┌─────────┐         ╱─────────╲              ┌716
│  PASS   │  PASS  ╱   CHECK    ╲    DROP    │  DROP   │
│ PACKET  │───────⟨GLOBAL POST-RULE⟩────────│ PACKET  │
└─────────┘        ╲    BASE?    ╱            └─────────┘
 722                ╲─────────╱
                         │ NO MATCH
          717    ╱─────────╲              723
                ╱  MATCHES   ╲    YES    ┌──────────┐
               ⟨   DEFAULT    ⟩─────────│ CARRY OUT │
                ╲   RULE?     ╱          │ DEFAULT   │
                 ╲─────────╱             │ ACTION    │
                     │ NO                 └──────────┘
          724 ─┌───────┐
               │ ERROR │
               └───────┘
```

FIG. 8A

FIG. 8B

```
A:  A─►✱  PASS        B:  B─►G  PASS        C:  ✱─►C  PASS
    G─►A  DROP            B─►H  DROP            C─►G  DROP
    H─►A  PASS            H─►B  PASS            C─►H  PASS
```

FIG. 8C

```
┌─────────────────────────────────────────────┐
│  h(A): 32      h(B): 61      h(C): 93        │──823
└─────────────────────────────────────────────┘

        LOCAL-IN                 LOCAL-OUT
                                                    ──822
821──  32:  G─►A  DROP       32:  A─►✱  PASS
            H─►A  PASS

       61:  H─►B  PASS       61:  B─►G  PASS
                                  B─►H  DROP

       93:  ✱─►C  PASS       93:  C─►G  DROP
                                  C─►H  PASS

┌─────────────────────────────────────────────┐
│  PACKET:  SOURCE    DESTINATION              │──824
│              H           B                   │
│           h(H)=88    h(B)=61                 │
└─────────────────────────────────────────────┘
```

FIG. 8D

841

CHECK PEER-IN HASH TABLE? → DROP → DROP PACKET (842)

PASS OR NO MATCH

843 — CHECK PEER-OUT HASH TABLE? → DROP → DROP PACKET (844)

PASS OR NO MATCH

BOTH PEER-IN AND PEER-OUT NO MATCH? → NO → PASS (845)

YES

846 — NO MATCH

FIG. 9

```
┌─────────────────┐
│      PEER       │ ─ 91
│ AUTHENTICATION  │
└─────────────────┘
         │
         ▼
┌─────────────────┐
│   PEER RULE     │ ─ 92
│  BASE LOADED    │
│  INTO FILTER    │
└─────────────────┘
         │
         ▼
┌─────────────────┐
│   CARRY OUT     │ ─ 93
│ HASH FUNCTION   │
│    ON PEER      │
│ NETWORK ADDRESS │
└─────────────────┘
         │
         ▼
┌─────────────────┐
│     UPDATE      │ ─ 94
│  PEER-IN AND    │
│    PEER-OUT     │
│  HASH TABLES    │
└─────────────────┘
         │
         ▼
┌─────────────────┐
│   PEER LOSES    │ ─ 95
│ AUTHENTICATION  │
└─────────────────┘
         │
         ▼
┌─────────────────┐
│  EJECT PEER'S   │ ─ 96
│   RULES FROM    │
│ FILTER'S LOCAL  │
│   RULE BASE     │
└─────────────────┘
         │
         ▼
┌─────────────────┐
│  EJECT PEER'S   │ ─ 97
│   POINTERS IN   │
│ PEER-IN AND -OUT│
│  HASH TABLES    │
└─────────────────┘
```

EUROPEAN SEARCH REPORT

European Patent Office

Application Number
EP 98 10 0283

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
X	WO 96 05549 A (SHIVA CORP)	1,4,5, 11,14, 15,24,26	H04L29/06 G06F1/00
	* page 3, line 8 - page 5, line 7 * * page 8, line 9 - page 9, line 32 *		
Y	* page 10, line 23 - page 11, line 2 * * page 12, line 28 - page 13, line 24 * * page 14, line 31 - page 15, line 21 * ---	3,13	
Y	BELLOVIN S M ET AL: "NETWORK FIREWALLS" IEEE COMMUNICATIONS MAGAZINE, vol. 32, no. 9, 1 September 1994, pages 50-57, XP000476555	3,13	
A	* page 52, column 1, line 60 - page 54, column 2, line 30 *	7,9,10, 12, 18-20,22	
A	US 5 473 607 A (HAUSMAN RICHARD J ET AL)	9,10,20, 22	
	* page 1, line 33 - line 45 * * column 2, line 47 - line 66 * * column 5, line 23 - line 40 * * column 9, line 35 - line 46 * ---		TECHNICAL FIELDS SEARCHED (Int.Cl.6) H04L G06F
P,X	EP 0 762 707 A (TELIA AB)	1-5, 11-15, 24,26	
	* page 2, column 1, line 34 - page 3, column 3, line 52 * -----		

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	21 April 1998	Karavassilis, N

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document